Project Title: 3D Thermal Evolution of Mercury

1. What application code or CS project did you run?

CitcomS is a 'community' finite element code for solving the equations of motion appropriate to creeping flow in the interior of a solid planet in a 3D spherical shell. (see for example, http://earth.leeds.ac.uk/dynamicearth/convection/) CitcomS uses a multi-grid solver and a UZAWA pressure scheme (see for example, http://www.mcc.monash.edu/twiki/view/Research/UzawaIterationProcedure) to solve Stokes equation. CitcomS was developed by a number of students and postdocs in geophysics and it now being maintained and developed by an NSF funded software development project. (see http://www.geodynamics.org)

2. What were your goals?

CitComS has never been run on more than about 400 processors (and that was by me at the BGL/ workshop at Argonne in April). The domain decomposition for the spherical shell breaks the shell into 12 caps, so it does not neatly map onto a power of 2 number of processors.

I was unsure whether CitComS could be mapped to the bluegene efficiently and whether it would really run on such a large number of processors. (It can and it does.) I hoped to get a computation, suitable to understanding the thermal evolution of the planet Mercury to run. This did seem to run successfully for a period of about 900 Million model years (about 1/6th the age of the solar system.)

3. What results did you achieve?

The 16-rack job ran for 2+ hours and produced more than 250,000 result files. (Each processor writes its part of the sphere to a local file, assuming each processor has a local scratch disk.) We have been scratching our heads as to how to get these files from Watson to Purdue.

To make use of 16k processors, we divided each of the 12 spherical caps into 11x11x11 processor domains, each with 8x8x8 node points. This means we used 15972 processors. Each iteration took about 0.6 seconds (on average).

Scott King
Purdue University